

## INTEGRATED MODELING THROUGH SYMBOLIC MANIPULATION

### Field of the Invention

The present invention relates to process modeling. More specifically, the present invention relates to methods and systems for producing integrated process models.

### Background of the Invention

Businesses are constantly driven by ever stronger global competitive pressures to reduce costs while complying with ever more stringent safety and environmental regulations. Many of these businesses are in industries that are heavily process based, such as the chemical industry, the petrochemical industry, and other manufacturing industries. Process modeling has been used to accelerate and enable design, testing, and implementation of new processes as well as improvements to existing processes. A broad variety of applications have found need for process modeling, including process simulation, optimization, advanced control, planning, scheduling, training, and maintenance. Equally numerous are the types of process models used by these applications: first principles, heuristic, rule-based, and empirical are examples.

However, model-based applications in process industries have not achieved their full potential since process models, which lie at the heart of those applications, are not efficiently used. The costs associated with developing, implementing, and maintaining process models often inhibit the efficient use of model-based process applications. There exist at least three major obstacles to the reduction of these costs: 1) lack of a model development methodology for creating and maintaining application-independent models, 2) lack of a framework for addressing model robustness and uncertainty, and 3) lack of an integrated software environment centered around plant-wide information databases.

Typically, process models are developed with methodologies and procedures tailored specifically for individual applications. Consequently, porting process models to new application environments invariably lead to duplication of effort and inconsistent results. Process engineers may spend significant amounts of time either developing the models from scratch or manually transferring and modifying process models from one

application to another. The time spent on such development and transfer is not only unproductive, but also error-prone as incorrect transfer and use frequently introduce errors. To cite an example of the problems mentioned above due to modeling inefficiencies, plant-wide optimization applications in process industries are frequently shut down because the underlying process model could not be easily updated to reflect changes in the plant. In summary, lack of efficient modeling mechanisms affects the performance of the model-based process applications, at the heart of which are process models.

### Summary of the Invention

An open, flexible, and integrated environment is used for process model development. Integrated modeling and symbolic manipulation is leveraged for the derivation, construction, maintenance, and reuse of application-independent models. Related models created for different applications share a common ancestry and maintain model consistency while enabling the models to share information about the process. The present invention finds utility in a broad variety of applications such as process design and analysis, process operations, maintenance, and logistics.

In one embodiment, environment independent proper ancestor models (PAMs) are created for generic components of a process, such as a distillation or flash column. An application may be comprised of multiple such processes corresponding to many different physical components. PAMs contain symbolic representations of different sub-processes that occur within a process or component. A user makes assumptions about the component, by indicating how sub-processes are to be considered or not considered. PAMs are then modified in accordance with the assumptions to derive a specific environment model (SEM). Conversion of multiple PAMs is easily performed to create a specific model for one application. The invention facilitates better inter-application consistency, plant-model consistency, and model reuse.

### Description of the Drawings

FIG. 1 illustrates a prior art set of equations for generically modeling a flash column

FIG. 2 illustrates a symbolic representation of the equations of FIG. 1.

FIG. 3 illustrates selected process assumptions for a particular flash column.

FIG. 4 illustrates further selected process assumptions for a particular flash column.

FIG. 5 illustrates still further selected process assumptions for a particular flash column.

FIG. 6 illustrates a set of equations representing a specific model for a flash column based on selected assumptions.

FIG. 7 illustrates a symbolic representation of the equations of FIG. 6.

FIG. 8 is a block diagram showing the use of proper ancestor models in generating specific environment models for multiple applications used in a manufacturing process.

FIG. 9 is a screen shot of an interface for specification of assumptions for generating a specific environment model for a distillation column.

FIG. 10 is a screen shot showing the specific environment model formed based on the specification of assumptions of FIG. 9.

### Detailed Description of the Invention

In the following detailed description of the present invention, reference is made to the accompanying Drawings, which form a part hereof, and in which are shown by way of illustration specific embodiments in which the present invention may be practiced. It should be understood that other embodiments may be utilized and structural changes may be made without departing from the scope of the present invention.

Proper ancestor models (“PAMs”) contain environment independent symbolic descriptions of various components of a manufacturing process. For example, in an embodiment useful to the chemical industry, multiple PAMs comprising a PAM library contain process models representing distillation columns, reactors, condensers, and heat exchangers of various designs, referred to as components. PAMs are generic in nature, and not influenced by environmental conditions. A specific environment model (SEM) representing a particular component is derived from a PAM by introducing additional assumptions describing the specifics of its physical implementation in the process or desired level of modeling abstraction. Listing all the assumptions explicitly and keeping

them in one place ensures that the same component is consistently described across many applications, even though each type of component may be represented (by SEMs) on different levels of abstraction in different applications.

A PAM is a description of a process containing no environment-specific  
5 knowledge. PAMs are written in a formal language (that is, a symbolic or equation-based format) with capabilities for the clear and natural description of mathematical expressions and constructs such as symbolic placeholders for unspecified operations or calculations. One example of a set of equations describing the behavior of a component such as a flash column in a typical process automation environment is shown in FIG. 1. FIG. 2 is a  
10 symbolic language representation of the same equations. Symbolic languages utilized in various embodiments comprise Mathematica (available from Wolfram Research), Axiom (available from Numerical Algorithms Group), MAPLE (available from Maple in Canada) and ADIFOR (automatic differentiation of Fortran) (available from Rice University), and possibly others currently existing and invented in the future. The  
15 Mathematica symbolic language representation is illustrated in FIG. 2.

Proper ancestor models are application independent. In order to be so, they focus on capturing the physical and chemical aspects of the modeled component's behavior, while avoiding all the aspects related to its particular implementation on the plant floor. Also specifics related to program implementation of the component model are avoided in  
20 PAMs to allow their use in and porting between applications written in different programming languages and using different programming methods. According to the present invention, process model developers (for example, engineers, developers, and plant operators) can focus on specific process issues while distinguishing process issues from specific environmental issues such as implementation platform and programming  
25 language. After a process model developer creates one or more PAMs, the PAMs are stored in a library. Thus, a set of proper ancestor models in an embodiment of the present invention is referred to as the PAM library. The PAM library is implementable on a single computer using a single database, for example. Alternately, the PAM library can be distributed over a network with different PAMs residing on different computers  
30 connected via a network.

Symbolic manipulation is used to construct SEMs from a library of PAMs in one embodiment. A user creates SEMs by choosing one or more PAMs from a PAM library and specifying process assumptions to customize the PAM into a desired SEM. The process assumptions embody more specific knowledge about the user-desired process model. The system combines the user-chosen assumptions with the user-chosen PAMs to produce the SEMs. The user-chosen assumptions could be one or more of the following. The process assumptions allow the user to add specific knowledge concerning the component's physical realization to a more generic process model (that is, a PAM). The methodological assumptions allow the user to choose specific mathematical methods and algorithms he prefers to solve the models. The programming assumptions allow the user to choose specific programming methods and languages, in which the SEMs are to be produced.. The construction of SEMs in accordance with the present invention ensures increased consistency across applications, better plant-model consistency, facilitation of model reuse and avoidance of duplication of effort in application development.

A user does not have to understand or know what language the process models or assumptions are written in. The present system presents symbolic representations of the process models and process assumptions to the user. The user chooses process models and process assumptions by choosing their symbolic representations. In one embodiment of the present invention, the user chooses process models and process assumptions from a menu of their symbolic representations displayed on a computer screen.

The equations illustrated in FIG. 1 constitute an embodiment of a PAM according to the present invention. FIG. 1 illustrates equations that describe a process typically embodied in a flash column, the simplest of process unit operations. Flash columns are used in the chemical industry to separate mixtures. For example, a distillation column could be considered as a stack of flash columns. The equations in FIG. 1 are typically created by an engineer wishing to describe or model a generic process found in a flash column. The equations in FIG. 1 by themselves are prior art, and are shown here only for illustrative purposes and simplicity. They can be found in the literature and are very fundamental.. For example, in FIG. 1, the variable  $M_V$  represents the amount of vapor present in the flash column, the variable  $M_L$  represents the amount of liquid present in the

flash column, and the variable  $P^*$  represents the vapor pressure. Once the equations are created, they can then be stored in the PAM library for future use.

The equations of FIG. 1 represent multiple aspects of the type of component it is modeling. At 110, the rate of change of the mass of vapor holdup is represented. Other representations include rate of change of the mass of liquid holdup 115, rate of change of the enthalpy of the vapor 120, enthalpy change of the liquid 125, mass transfer correlation 130, energy transfer correlation 135, vapor and liquid enthalpy equations 140 and 145, vapor pressure equation 150, gas law 155 and volume correlation 160.

FIG. 2 illustrates another embodiment of a PAM according to the present invention. Conceptually, the equations in FIG. 2 define the same model of a flash column. However, viewed as a description of process knowledge, the model in FIG. 2 is different from the model in FIG. 1. Rather than being recorded in the traditional mathematical notation which computers cannot safely understand due to its ambiguity, the equations illustrated in FIG. 2 are written in a symbolic format using Mathematica. A user desiring to create a PAM according to the present invention can use Mathematica as the formal language for expressing the equations that comprise the PAM.

A user wishing to create a specific process model first chooses one or more PAMs, such as the PAM of a flash column illustrated in FIG. 1, and then gradually adds desired process methodological and programming assumptions in that order in one embodiment. These assumptions embody more specific knowledge about the desired process to be modeled and the way it should be modeled. For example, FIG. 3 illustrates equations representing the process assumption that a process model, such as the one embodied in FIG. 1, exhibits thermal equilibrium. Similarly, FIG. 4 illustrates equations representing an assumption of material equilibrium and FIG. 5 illustrates equations representing the assumption that there exists no vapor holdup.

The usual or more common process assumptions, as well as the PAMs are chosen by use of a graphical interface that provides such options in common language such as English in one embodiment. That is the user does not have to enter the equations into the system. In fact, a user is not even required to know any implementation details about the PAMs or the process assumptions. The user is not required to know what the underlying equations look like or even what formal language they are written in. For example, in

one embodiment the user simply chooses the process assumptions by choosing individual process assumptions from a menu of process assumptions displayed on a computer screen. In addition to that, a more sophisticated user can enter atypical or unique assumptions directly in the symbolic language used to encode PAMs.

5           A user of the present invention may manipulate the equations defining PAMs to create equations defining more specific models without having to know any of the underlying equations or even what formal language the equations were written in. The user chooses assumptions (for example, thermal equilibrium) and the system then manipulates the underlying symbolic equations to produce the desired model for the user.  
10   Again, if desired, a more sophisticated user can direct the PAM customization process by specifying methodological and programming assumptions in a suitable language.

          Once a user has chosen at least one PAM and at least one process assumption, the present system then combines the user-chosen process assumption(s) with the user-chosen PAM(s) to produce one or more specific process models. This is accomplished  
15   by merging the equations representing the user-chosen assumption(s) with the equations representing the user-chosen PAM(s) to produce a new set of equations, which represent a specific process model. Thus, each specific process model is represented by a set of equations produced by combining equations from one or more PAMs with the equations representing one or more process assumption(s). An SEM Builder combines PAMs with  
20   process assumptions to produce specific process models.

          In this manner, an SEM can be thought of as a model for a particular application derived by a rigorous and well-documented process from one or more PAMs residing in the PAM library through its assumption-driven customization. Allowing the user to symbolically choose one or more PAMs and choose one or more process assumptions  
25   enables the extraction of fidelity levels appropriate for each application. For example, process design calculations generally involve geometric and capacity considerations, while process control applications are generally involved with adjusting operating conditions and variables to satisfy certain constraints.

          FIG.s 6 and 7 illustrate two examples of equations representing specific process  
30   models. FIG. 6 illustrates equations produced by combining the PAM of FIG. 1 with assumptions of thermal equilibrium, material equilibrium, and no vapor holdup. FIG. 7

illustrates a symbolic representation produced by combining the PAM of FIG. 2 with these same assumptions.

Specific environment models of the present invention can be developed all the way to computer programming code that is executable on a proposed platform. The use  
5 of a symbolic language enables autogeneration of code directly from the symbolic representation of the PAM given the assumptions in a known manner. For example, the derived SEM is a computer-implemented simulation of the underlying specific process model.

For each SEM created, the SEM Builder creates a transformation log. The  
10 transformation log contains information necessary to record how the SEM was created. This information includes, for example, what PAMs were chosen and what assumptions were chosen as well as how the underlying equations were combined. The transformation log can be thought of as a history of how a SEM was derived. The transformation log helps to identify the contents of a model and facilitates the model's  
15 reuse. For example, the transformation log allows a user to start with a previously created SEM for a similar application and work backwards to determine what if any assumptions need to be changed to produce a new SEM more suitable for a new application.

In one embodiment, the present invention is a computer-implemented method.  
20 The computer programming languages that can be utilized to implement the methods of the present invention are not particularly critical. For example, methods of the present invention could be implemented using the C programming language, C++, Visual Basic, etc. Methods according to the present invention may be implemented so as to run on a single computer or on multiple computers connected via a network.

25 The computing hardware utilized to perform methods of the present invention or utilized to create system according to the present invention is not particularly critical. For example, in one embodiment the present invention is a single computer containing computer-executable instructions capable of performing the methods according to the present invention. The particular type of computer can be determined by a practitioner of  
30 ordinary skill in the programming arts. Alternatively, the present invention can be embodied in multiple computers connected via a network. The multiple computers



may vary in power and type and may vary in what parts of the present invention they embody. For example, the PAM library may exist on a server connected to user workstations via a network. The network can be a local area network or the Internet, for example. The user workstations provide the various users the ability to interact with the system. For example, a process designer can interact with one workstation to create PAMs, which get sent over the network and are stored in the PAM library residing on the server. A user desiring a specific process model can work at a second workstation, choosing appropriate PAMs and entering process and other assumptions. The second workstation retrieves the appropriate PAMs over the network from the PAM library and combines them with the user-entered process assumptions to produce one or more SEMs.

FIG. 8 is a block diagram showing the use of proper ancestor models in generating specific environment models for multiple applications used in a manufacturing process. It illustrates the overall process of using PAMs to create SEMs for modeling a manufacturing process. Each PAM is used to create one or more SEMs as shown by modifying the PAM based on sets of assumptions. Several different types of SEMs are shown each corresponding to one or more applications. They include but are not limited to process design, process economy, process operations, process control and process retrofit. Many others may also be utilized.

FIG. 9 is a screen shot of an interface for specification of assumptions for generating a specific environment model for a distillation column. The distillation column is first identified, followed by specific assumptions being selected, such as by a user typing in a logical value, or selecting from a menu or pull down menu. For instance, some of the selections for this distillation column include presence of vapor hold up, and steady state conditions.

FIG. 10 is a screen shot showing the specific environment model formed based on the specification of assumptions of FIG. 9. The assumptions are applied to the symbolic representations of a PAM to generate an actual specific environment model. The model reflects the logical choices made in FIG. 9, and reflect at least vapor hold up and steady state conditions in the SEM that is generated.

The present invention provides significant improvements in process modeling. For example, the present invention helps integrate model development, ensure plant-

model consistency, manage model uncertainty, facilitate model reuse, support the model-based needs of a variety of clients, and reduce costs of model-based applications. The present invention provides these improvements by utilizing integrated modeling and symbolic manipulation concepts embodied in proper ancestor models (“PAMs”) and  
5 specific environment models (“SEMs”).